Binary Partition Trees-based spectral-spatial permutation ordering

Miguel Angel Veganzones, Mauro Dalla Mura, Guillaume Tochon, Jocelyn Chanussot

Binary Partition Trees-based spectral-spatial permutation ordering

Miguel A. Veganzones[1], Mauro Dalla Mura[2], Guillaume Tochon[2], and Jocelyn Chanussot[2,3]

[1] GIPSA-lab, CNRS, Saint Martin d'Hères, France
miguel-angel.veganzones@gipsa-lab.fr
[2] GIPSA-lab, Grenoble-INP, Saint Martin d'Hères, France
[3] Faculty of Electrical and Computer Engineering, University of Iceland, Reykjavik, Iceland

Abstract. Mathematical Morphology (MM) is founded on the mathematical branch of Lattice Theory. Morphological operations can be described as mappings between complete lattices, and complete lattices are a type of partially-ordered sets (*poset*). Thus, the most elementary requirement to define morphological operators on a data domain is to establish an ordering of the data. MM has been very successful defining image operators and filters for binary and gray-scale images, where it can take advantage of the natural ordering of the sets $\{0, 1\}$ and $\mathbb{Z}^+$. For multivariate data, *i.e.* RGB or hyperspectral images, other orderings such as reduced orderings (R-orderings) have been proposed. Anyway, all these orderings are based solely on sorting the spectral set of values, that is finding a useful permutation of the data samples. Here, we propose to define an ordering based on both, the spectral and the spatial information, by means of a binary partition tree (BPT) representation of images. This novel permutation ordering takes into account not only the possible spectral ordering but also the hierarchies of spatial structures encoded in the BPT.

Keywords: Mathematical Morphology, Lattice Theory, Ordering, Spectral Spatial analysis, Binary Partition Trees, Permutations

1 Introduction

Mathematical Morphology (MM) [15, 10, 16, 7, 18] has been very successful defining image operators and filters for binary and gray-scale images. A very appealing characteristic of MM is that its mathematical support is well known. Lattice Theory [5, 8, 9] gives the most general theoretical background for MM [13, 11]. Basically, morphological operations can be described as mappings between complete lattices.

A binary relation ϱ satisfying reflexivity, antisymmetry and transitivity properties is called an *order,* denoted by $\leq$. A non-empty set P endowed with an order relation, $\langle P; \leq \rangle$, is a *partially-ordered set* or *poset*, denoted by $\mathcal{P}$. A poset

$\mathcal{L} = \langle L; \leq \rangle$ is a *lattice* if an infimum and a supremum exist for any pair of elements of L. Formally, a poset $\mathcal{L} = \langle L; \leq \rangle$ is a lattice iff $\inf H$ and $\sup H$ exist for any finite non-empty subset $H \subseteq L$. The lattice Duality Principle states that if $\mathcal{L} = \langle L; \leq \rangle$ is a lattice, the dual $\mathcal{L}^\delta = \langle L; \geq \rangle$ is a lattice too. A lattice can also be described in an algebraic form by setting $a \wedge b = \inf \{a, b\}$ and $a \vee b = \sup \{a, b\}$. Thus, the algebra $\langle L, \wedge, \vee \rangle$ is equivalent to the lattice $\mathcal{L}$, $\langle L, \wedge, \vee \rangle \equiv \langle L; \leq \rangle$. In MM, operators are defined over an important type of lattices, the *complete lattices.* A lattice $\mathcal{L} = \langle L; \leq \rangle$ is complete iff $\inf H$ and $\sup H$ exist for any subset $H \subseteq L$. A complete lattice has both a smallest element called *bottom*, denoted as $\perp$, and a greatest element called top, denoted as $\top$. All finite lattices are complete lattices. From now on, we denote complete lattices by the symbols $\mathbb{L}$ and $\mathbb{M}$.

For every subset $Y \subseteq \mathbb{L}$ an *erosion* is a mapping $\varepsilon : \mathbb{L} \to \mathbb{M}$ that commutes with the infimum operation, $\varepsilon \left(\bigwedge Y \right) = \bigwedge_{y \in Y} \varepsilon(y)$. Similarly, a *dilation* is a mapping $\delta : \mathbb{L} \to \mathbb{M}$ that commutes with the supremum operation, $\delta \left(\bigvee Y \right) = \bigvee_{y \in Y} \delta(y)$. An *anti-erosion* operator, $\bar{\varepsilon} : \mathbb{L} \to \mathbb{M}$, and an *anti-dilation* operator, $\bar{\delta} : \mathbb{L} \to \mathbb{M}$, are defined as mappings holding $\bar{\varepsilon} \left(\bigwedge Y \right) = \bigvee_{y \in Y} \bar{\varepsilon}(y)$ and $\bar{\delta} \left(\bigvee Y \right) = \bigwedge_{y \in Y} \bar{\delta}(y)$, respectively. Any mapping Ψ between complete lattices $\mathbb{L}$ and $\mathbb{M}$ can be expressed in terms of supremums and infimums of these four morphological operators [3].

Binary and gray scale images are complete lattices given the natural order of the binary set, $\{0, 1\}$, and the positive integers, $\mathbb{Z}^+$, respectively (or the set of real vectors, $\mathbb{R}$, in general). The extension of MM to multivariate images is not straightforward since pixels are (high dimensional) vectors without an intrinsic natural total order. There are different strategies to build up an order from multivariate data [4, 12, 17, 2]. The Marginal ordering (M-ordering) corresponds to univariate orderings realized on every component of the given vectors:

$$\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n,\ \mathbf{x} \leq \mathbf{y} \Longleftrightarrow x_i \leq y_i,\ \forall i \in \{1, \ldots, n\} \tag{1}$$

The multivalued ordering is given by the independent ordering of each vector component, so the M-ordering is also called *component-wise ordering.* The M-ordering allows the use of MM but all the between components information is ignored. Furthermore, the use of a M-ordering results in the apparition of vectors which are not present in the original image, what is called the *false color problem.* Another common approach is the Conditional ordering (C-ordering) which establishes a priority between the vectors marginal components. The vectors components are ranked and sequentially selected according to this rank. *Lexicographical ordering* is the most known example of C-ordering:

$$\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n,\ \mathbf{x} \leq \mathbf{y} \Longleftrightarrow \exists i \in \{1, \ldots, n\},\ (\forall j < i,\ x_j = y_j) \wedge (x_i \leq y_i) \tag{2}$$

The C-ordering is useful when a natural priority exists among vector components, which is not often the case of multi and hyperspectral images. Even when this is suitable, the use of C-orderings as the lexicographical ordering, yields to the use of only a few components dismissing the information contained in the ones left.

One way to define a multivariate ordering that makes use of all the components and interdependencies is by constructing a surjective mapping into a lattice $h : \mathbb{R}^n \to \mathbb{L}$ so that we can assume an ordering induced by this mapping. The h-ordering, $\leq_h$, is defined as:

$$\mathbf{x} \leq_h \mathbf{y} \Leftrightarrow h(\mathbf{x}) \leq h(\mathbf{y})\,; \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n \tag{3}$$

Some authors [1, 23] define h-orderings on the basis of a supervised classifier trained with some pixel values. Discriminant function values and class a posteriori probabilities provide the surjective mapping h.

All the previous orderings, that allow to define complete lattices, and thereof, to apply mathematical operators over binary, gray-scale or multivariate images, are based on the ordering of the set of spectral values. Here, we propose to define an ordering using not only the spectral information but also the spatial location of the data vectors, and in particular, the spatial structures encoded in a binary partition tree (BPT) representation of images [14]. We show the potential use of this novel BPT-based spectral-spatial permutation ordering, and we discuss on further extensions of the proposed ideas.

2 Binary Partition Trees

Hierarchical segmentation algorithms have proved to be very valuable to explore and exploit the spatial content of images by providing a hierarchy of segmentations working at different scales. The BPT is a hierarchical region-based representation of an image in a tree structure [14]. Recently, some authors have proposed the use of the Binary Partition Tree (BPT) to handle very high dimensional images such as hyperspectral images [20, 21, 19, 22].

In the BPT representation, the leaf nodes correspond to an initial partition of the image, which can be the individual pixels, or a coarser segmentation map. From this initial partition, an iterative bottom-up region merging algorithm is applied until only one region remains. This last region represents the whole image and corresponds to the root node. All the nodes between the leaves and the root result of the merging of two adjacent children regions. An example of BPT is displayed in Fig. 1. If the initial partition contains n leaf nodes, the BPT representation contains $2n - 1$ nodes.

Two notions are of prime importance when defining a BPT: i) the *region model* $\mathcal{M}_{\mathcal{R}}$ which specifies how a region $\mathcal{R}$ is modelled, and ii) the *merging criterion*, $\mathcal{O}(\mathcal{M}_{\mathcal{R}_\alpha}, \mathcal{M}_{\mathcal{R}_\beta})$, which is a dissimilarity measure between the region models of any two regions $\mathcal{R}_\alpha$ and $\mathcal{R}_\beta$. Each merging iteration involves the search of the two neighbouring regions which achieve the lowest pair-wise dissimilarity among all the pairs of neighbouring regions in the current segmentation map. Those two regions are consequently merged.

Given a hyperspectral region $\mathcal{R}$, with $N_{\mathcal{R}}$ hyperspectral samples $\mathbf{r}_j \in \mathbb{R}^q$, $j \in 1 \ldots N_{\mathcal{R}}$, the first-order parametric model $\mathcal{M}_{\mathcal{R}}$ is defined by the average of

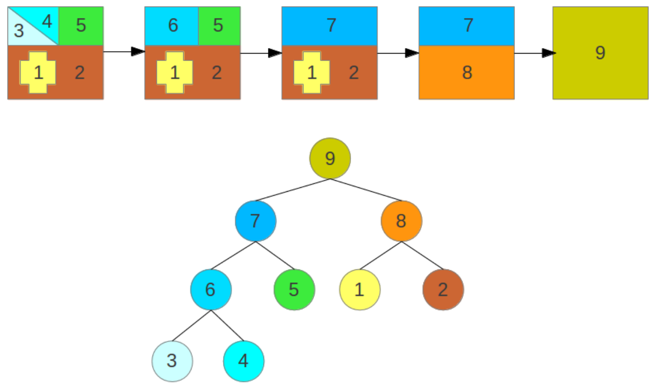


Fig. 1. Construction of the Binary Partition Tree (BPT).

the hyperspectral samples $\bar{\mathbf{r}}$ in each band $k = 1, \ldots, q$:

$$\mathcal{M}_{\mathcal{R}}{}^{(k)} \stackrel{d}{=} \bar{r}^{(k)} = \frac{1}{N_{\mathcal{R}}} \sum_{j=1}^{N_{\mathcal{R}}} r_j^{(k)}. \tag{4}$$

Using the first-order parametric model (4), a merging criterion is defined as the Euclidean, d_{EUC}, or the spectral angle distance, d_{SAD}, between the average values of any two adjacent regions:

$$\mathcal{O}_{\mathrm{EUC}}\left(\mathcal{M}_{\mathcal{R}_\alpha}, \mathcal{M}_{\mathcal{R}_\beta}\right) \stackrel{d}{=} d_{\mathrm{EUC}}\left(\bar{\mathbf{r}}_\alpha, \bar{\mathbf{r}}_\beta\right) = \sqrt{\sum_{k=1}^{q} \left(\bar{r}_\alpha^{(k)} - \bar{r}_\beta^{(k)}\right)^2}, \tag{5}$$

$$\mathcal{O}_{\mathrm{SAM}}\left(\mathcal{M}_{\mathcal{R}_\alpha}, \mathcal{M}_{\mathcal{R}_\beta}\right) \stackrel{d}{=} d_{\mathrm{SAD}}\left(\bar{\mathbf{r}}_\alpha, \bar{\mathbf{r}}_\beta\right) = \arccos\left(\frac{\bar{\mathbf{r}}_\alpha^T \bar{\mathbf{r}}_\beta}{\|\bar{\mathbf{r}}_\alpha\| \|\bar{\mathbf{r}}_\beta\|}\right). \tag{6}$$

The building of a BPT may suffer from small and meaningless regions resulting in a spatially unbalanced tree. To overcome this limitation, a priority term is included in the merging criterion that forces those regions smaller than a given percentage of the average region size to be merged first [6, 19].

3 BPT-based spectral-spatial permutation ordering

Given a BPT representation of an image, the leaves are sorted from left to right following the hierarchical structure in an arbitrary way, normally selected according to implementation aspects. In other words, in the construction of the tree there is no ordering between sibling nodes. In this work, we propose to impose an ordering among nodes in the tree. Specifically, the ordering is done on the leaves of the tree. Given an image with n leaf nodes, *i.e.* with n pixels, there are $n-1$ merging operations need to build the BPT. Each of the merging operations implies a decision about the sorting of nodes that are being merged. Thus, there are 2^{n-1} possible permutations of the BPT leaves. Next, we propose a criterion to select one among all the possible permutations of the leaves using the hierarchical spatial structures encoded in the BPT representation and to use such permutation to define a BPT-based spectral-spatial permutation ordering of the image pixels.

3.1 BPT sorting

Let $\boldsymbol{\Pi} = \left\{\boldsymbol{\pi}^{(k)}\right\}_{k=1}^{2^{n-1}}$ be the set of all the possible permutations of the leaves of a given BPT representation of an image, where each $\boldsymbol{\pi}^{(k)} = \left[\pi_1^{(k)}, \ldots, \pi_n^{(k)}\right]$, denotes a permutation that sorts the n leaves from left-to-right (see Fig. 2).

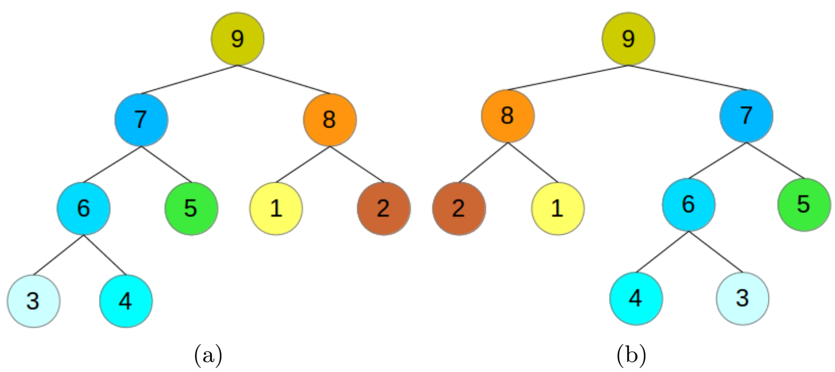


Fig. 2. Two examples of possible random sortings of the BPT leaves: (a) $\boldsymbol{\pi}^1 = [4, 5, 1, 2, 3]$; and (b) $\boldsymbol{\pi}^2 = [2, 1, 4, 3, 5]$.

We propose as a criterion to select one among the set of possible sortings, $\boldsymbol{\Pi}$, of a given BPT, a top-down approach where at each node, the child node with minimum dissimilarity to the parent node is set to the left. The pseudo-code of this method that we have termed as the *minimum dissimilarity top-down BPT sorting* is given in Fig. 1. The proposed method sorts the BPT recursively starting from the root node. The recursion goes down by looking for the children of the node and calling himself to sort the sub-trees defined by each of the child nodes. The recursion stops when the node is a leaf node. Then, it goes back comparing the region models of the child nodes with respect to the region model of the parent node, setting to the left the sorted indexes of the child node with minimum region dissimilarity.

Algorithm 1 Minimum dissimilarity top-down BPT sorting pseudo-code

1. $\mathcal{T} \leftarrow \mathrm{BPT}(\mathbf{X})$ ▷ BPT construction
2. $\mathcal{R}_0 \leftarrow \mathrm{root}(\mathcal{T})$, ▷ Top-down initialization
3. $\boldsymbol{\pi} \leftarrow \mathrm{sortBPT}(\mathcal{R}_0)$ ▷ recursive sorting algorithm

```
function sortBPT(R)
    [R_c1, R_c2] ← children(R)            ▷ Children nodes
    if isLeaf(R_c1) then
        π1 = [ω(R_c1)]                    ▷ ω(·) returns node's index
    else
        π1 = sortBPT(R_c1)                ▷ Sort child node
    end
    if isLeaf(R_c2) then
        π2 = [ω(R_c2)]                    ▷ ω(·) returns node's index
    else
        π2 = sortBPT(R_c2)                ▷ Sort child node
    end
    if d(R, R_c1) ≤ d(R, R_c2) then       ▷ Compare region models
        π = [π1, π2]
    else
        π = [π2, π1]
    end
    return π
end
```

3.2 BPT-based spectral-spatial ordering

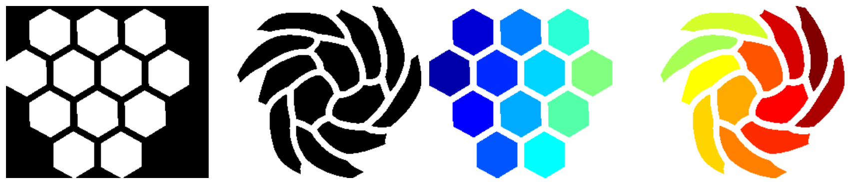
Once we have sorted the leaves of a given BPT representation of an image $\mathbf{X}$, that is, once we have selected a permutation $\boldsymbol{\pi}$ of the leaves, we propose to take advantage of it to define a permutation $\boldsymbol{\pi}$-ordering:

$$\mathbf{x} \leq_{\boldsymbol{\pi}} \mathbf{y} \Leftrightarrow \pi_{\omega(\mathbf{x})} \leq \pi_{\omega(\mathbf{y})};\ \forall \mathbf{x}, \mathbf{y} \in \mathbf{X}, \tag{7}$$

where $\omega(\cdot)$ is a function that given a pixel returns the index of the leaf containing it. Using the $\boldsymbol{\pi}$-ordering, an erosion (dilation) over a pixel will move pixel values toward the ones located in leaves on the left (right) of the sorted BPT defined by the permutation $\boldsymbol{\pi}$.

4 Examples

In order to show the potential use of the proposed minimum-dissimilarity top-down BPT sorting algorithm and the $\boldsymbol{\pi}$-ordering defined on base to the sorting of the BPT leaves, we have applied the proposed methodology to the logo of the ISMM conference in binary and RGB versions, and also on a grayscale and RGB image from the COREL dataset.



(a) (b)

Fig. 3. Application of the proposed BPT-based ordering methodology to the logo of the ISMM conference: (a) original binary logo, (b) original RGB logo.

4.1 ISMM logo

Fig. 3 shows the binary and RGB versions of the ISMM logo. In Fig. 4 we compare the result of applying erosion and dilation operations using a 11×11 square structural element, given a conventional component-wise ordering and the proposed $\boldsymbol{\pi}$-ordering from a BPT representation of the images obtained using the first-order parametric model (4) and the Euclidean distance (5). Figs. 4(a-d) show the results obtained by the conventional component-wise ordering, while Figs. 4(e-h) depicts the results obtained by the proposed $\boldsymbol{\pi}$-ordering.

For the binary logo, the conventional ordering results in shrinking the white areas and enlarging the black ones for the erosion operation, and the contrary for the dilation. However, the morphological operations using the proposed $\boldsymbol{\pi}$-ordering enlarge or reduce the geometrical structures of the logo independently of the foreground and background colors. That is, it works directly on the foreground and background structures independently of the binary encoding used to represent them in each case. For the RGB logo, we were interested in highlight that the proposed $\boldsymbol{\pi}$-ordering naturally operates with multivariate data. The conventional component-wise ordering can not handle the multivariate RGB information properly, modifying the colors of the geometrical structures, while the proposed $\boldsymbol{\pi}$-ordering is able to return the expected result.

4.2 Real image

Next, we provide a comparison using a real image from the COREL database. In all cases, we make use of a 11×11 square structural element. Fig. 5 depicts the gray-scale and RGB versions of the image. Fig. 6 depicts the results obtained using conventional gray-scale and RGB component-wise morphological operations compared to the proposed $\boldsymbol{\pi}$-ordering. The potential use of the proposed $\boldsymbol{\pi}$-ordering could be appreciated in the background. This is composed of sand with multiple randomly located dark and bright spots, probably corresponding to small rocks, salt grains and other small particles in the sand. Using conventional morphological operators it is not possible to get rid of both dark and bright structures, while using the proposed methodology, either both structures are enlarged or reduced. Here, we want to highlight more the potential

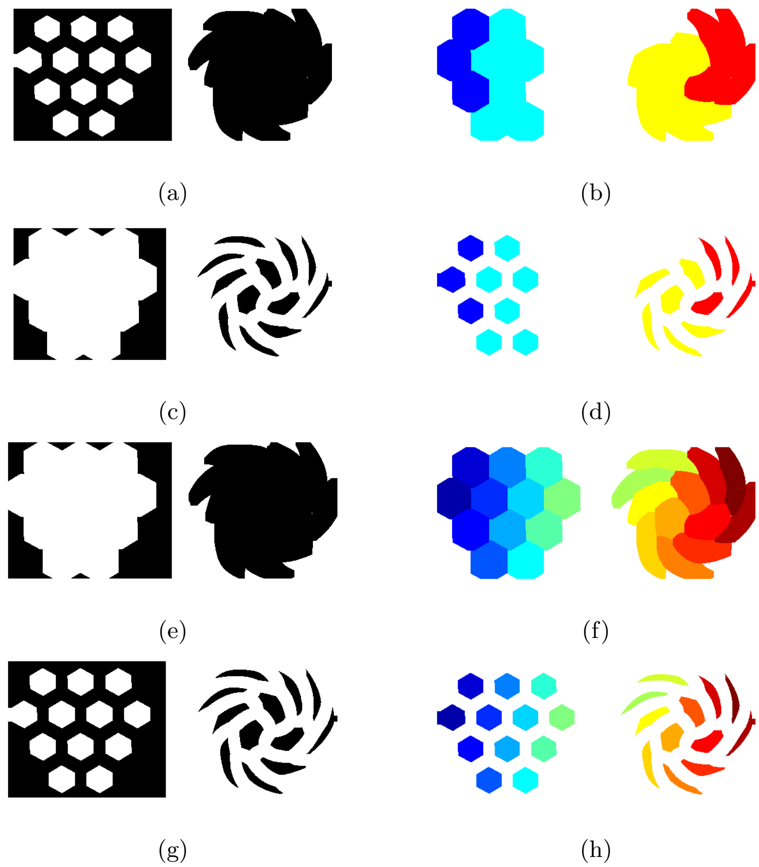


Fig. 4. Application of the erosion and dilation operations to the ISMM logos using a square 11×11 structural element: (a) conventional erosion of the binary logo, (b) component-wise erosion of the RGB logo, (c) conventional dilation of the binary logo, (d) component-wise dilation of the RGB logo, (e) proposed erosion of the binary logo, (f) proposed erosion of the RGB logo, (g) proposed dilation of the binary logo, and (h) proposed dilation of the RGB logo.

(a) (b)

Fig. 5. Real image obtained from the COREL database: (a) gray-scale image, and (b) RGB image.

of the proposed BPT-based spectral-spatial $\boldsymbol{\pi}$-ordering to work on the spatial structures encoded in the BPT representation.

5 Discussion

We would like to point-out some aspects and open discussions of the proposed methodology we are working on:

- We have proposed a definition of a permutation ordering, which could be understood as a kind of a reduced ordering, where the function $h(\cdot)$ corresponds to the permutation indexes. However, it could be also understood the other way around, where the reduced orderings are particular cases of a more general class of permutation orderings, where each proposed $h(\cdot)$ definition yields to a permutation of the data samples, and in particular, of the image pixels.
- We have also focused on a BPT-based definition of the proposed $\boldsymbol{\pi}$-ordering, taking advantage of the spectral-spatial representation of the image that the BPT methodology provides. However, the same underlined methodology could be defined for any structure that encodes a hierarchy of partitions, or that at least, in the case of trees, their leaves form a partition of the data.
- Finally, we have not be on time to provide results on which these authors think it could be a key point on the use of the proposed BPT-based spectral-spatial $\boldsymbol{\pi}$-ordering: the possibility of using the hierarchical structure of the tree to define structure elements based on sub-tree properties and to provide intuitive ways to define spatially selective mask images.

Fig. 6. Comparison of the conventional gray-scale (row 1) and the conventional RGB component-wise ordering (row 3) morphological operations, with respect to the proposed BPT-based spatial-spectral $\boldsymbol{\pi}$-ordering (rows 2 and 4) obtained using a square 11×11 structural element: (a) erosion, (b) dilation, (c) closing, (d) opening, (e) closing by reconstruction, and (f) opening by reconstruction.

6 Conclusions

We have proposed a novel spectral-spatial permutation ordering taking advantage of the hierarchical structure of the BPT representation of an image. The proposed $\boldsymbol{\pi}$-ordering, makes use of the sorted leaves of the BPT representation. We have also provided a top-down recursive algorithm to sort the BPT leaves. As far as the authors know, this is the first time that an ordering taking into account both, the spectral and the spatial information of the pixels of an image, is proposed. The proposed ordering allows to perform morphological operations that work on the spatial structures of the image encoded in the BPT representation. We have provided some examples of the potential use of the proposed ordering using binary, gray-scale and RGB images. It is worthy to note, that in addition to the novel spatial properties of the morphological operations, the proposed ordering naturally deals with multivariate data, *i.e.* RGB images, once a BPT representation of the image is given. Further work will explore the theoretical properties of the proposed ordering and potential new research avenues provided in the discussion section.

References

1. Angulo, J.: Morphological colour operators in totally ordered lattices based on distances: Application to image filtering, enhancement and analysis. Comput. Vis. Image Underst. 107(1-2), 56–73 (Jul 2007), http://dl.acm.org/citation.cfm?id=1265986.1266116
2. Aptoula, E., Lefevre, S.: A comparative study on multivariate mathematical morphology. Pattern Recogn. 40(11), 2914–2929 (Nov 2007), http://dl.acm.org/citation.cfm?id=1274191.1274319
3. Banon, G., Barrera, J.: Decomposition of mappings between complete lattices by mathematical morphology, part i. general lattices. Signal Processing 30(3), 299–327 (Feb 1993), http://www.sciencedirect.com/science/article/pii/0165168493900153
4. Barnett: The ordering of multivariate data. Journal Of The Royal Statistical Society Series A General 139(3), 318–355 (1976)
5. Birkhoff, G.: Lattice theory. AMS Bookstore (1995)
6. Calderero, F., Marques, F.: Region merging techniques using information theory statistical measures. IEEE Transactions on Image Processing 19(6), 1567–1586 (June 2010)
7. Goutsias, J., Heijmans, H.: Mathematical Morphology. IOS Press (Jan 2000)
8. Gratzer, G.: General Lattice Theory. Birkhauser Basel, 2 edn. (Jan 2003)
9. Gratzer, G.: Lattice Theory: Foundation. Springer Basel, 1st edition. edn. (Feb 2011)
10. Haralick, R., Sternberg, S., Zhuang, X.: Image analysis using mathematical morphology. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-9(4), 532–550 (Jul 1987)
11. Hawkes, P., Heijmans, H., Kazan, B.: Morphological Image Operators. Academic Press (Dec 1993)
12. Pitas, I., Tsakalides, P.: Multivariate ordering in color image filtering. IEEE Transactions on Circuits and Systems for Video Technology 1(3), 247–259, 295–6 (Sep 1991)

13. Ronse C.: Why mathematical morphology needs complete lattices. Signal Processing 21(2), 129–154 (Oct 1990)
14. Salembier, P., Garrido, L.: Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. IEEE Transactions on Image Processing 9(4), 561–576 (2000)
15. Serra, J.: Image Analysis and Mathematical Morphology, Volume 1 (Image Analysis & Mathematical Morphology Series). Academic Press (Feb 1984)
16. Serra, J.: Image Analysis and Mathematical Morphology, Vol. 2: Theoretical Advances. Academic Press, 1st edn. (Feb 1988)
17. Serra, J.: Anamorphoses and function lattices. In: Proceedings of SPIE. vol. 2030, pp. 2–11 (1993)
18. Soille, P.: Morphological Image Analysis: Principles and Applications. Springer, 2nd edn. (Jun 2004)
19. Tochon, G., Feret, J., Martin, R., Tupayachi, R., Chanussot, J., Asner, G.: Binary partition tree as a hyperspectral segmentation tool for tropical rainforests. In: 2012 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). pp. 6368–6371 (2012)
20. Valero, S., Salembier, P., Chanussot, J.: Comparison of merging orders and pruning strategies for binary partition tree in hyperspectral data. In: 2010 17th IEEE International Conference on Image Processing (ICIP). pp. 2565–2568. IEEE (2010)
21. Valero, S., Salembier, P., Chanussot, J.: Hyperspectral image representation and processing with binary partition trees. IEEE Transactions on Image Processing 22(4), 1430–1443 (2013)
22. Veganzones, M., Tochon, G., Dalla-Mura, M., Plaza, A., Chanussot, J.: Hyperspectral image segmentation using a new spectral unmixing-based binary partition tree representation. IEEE Transactions on Image Processing 23(8), 3574–3589 (Aug 2014)
23. Velasco-Forero, S., Angulo, J.: Supervised ordering in R^p: Application to morphological processing of hyperspectral images. IEEE Transactions on Image Processing 20(11), 3301–3308 (Nov 2011)